

DESCRIPTION

THREE-DIMENSIONAL SHAPE DRAWING DEVICE AND THREE-DIMENSIONAL
SHAPE DRAWING METHOD

5 TECHNICAL FIELD

[0001] The present invention relates to a three-dimensional
shape drawing device and a three-dimensional shape drawing method,
and more particularly to a three-dimensional shape drawing device
and a three-dimensional shape drawing method for drawing a
10 three-dimensional shape by using Z buffer algorithm using a depth
value from a viewpoint.

BACKGROUND ART

[0002] There have been three-dimensional shape displaying
15 devices for drawing a three-dimensional shape by using Z buffer
algorithm as a hidden surface removal system. Also, there have
been techniques for increasing a speed of drawing processing of
such a three-dimensional shape displaying device, for example,
a technique in which each pixel contains one bit flag memory (e.g.,
20 patent document 1) and a technique which uses a degenerated Z buffer
for, among a plurality of depth values of pixels, storing a depth
value of a farthest pixel from a view point and a depth value of
a nearest pixel from the view point (e.g., patent document 2).

[0003] FIG. 7 is a block diagram showing a configuration of
25 a three-dimensional shape drawing device 200 described in the

patent document 2. FIG. 8 is a diagram schematically showing a screen 214 for displaying a three-dimensional shape drawn by the three-dimensional shape drawing device 200 shown in FIG. 7 and a configuration of a ZR buffer 205.

5 [0004] In FIG. 7, an image supply section 208 supplies, to a pixel drawing section 209 and a drawing determination section 210 of an image composing section 201, data of a polygon (hereinafter, referred to as polygon data) to be drawn. The image composing section 201 converts the polygon data into pixel data.

10 [0005] As shown in FIG. 8, the screen 214 is divided into $M \times N$ blocks. A block 213 being a portion of a divided image signal comprises $K \times L$ pixels. The ZR buffer 205 retains a MAXZ and a MINZ of each of the blocks.

[0006] A Z main buffer 206 stores all Z values (depth values).

15 The ZR buffer 205 retains a maximum Z value (MAXZ) 211 and a minimum Z value (MINZ) 212 of each of the blocks shown in FIG. 8.

[0007] The pixel drawing section 209 outputs, to a drawing determination section 210, a Z value of a pixel to be drawn, and also outputs X and Y coordinates indicating a position of the pixel.

20 Based on either only the Z values retained by the ZR buffer 205 or both the Z values retained by the ZR buffer 205 and the Z values retained by the Z main buffer 206, the drawing determination section 210 determines whether or not to draw the pixel.

[0008] A frame buffer 203 retains the pixel which has been
25 determined, by the determination section 210, to be drawn. A

palette circuit 204 converts data of the pixel retained by the framebuffer 203 into brightness data indicating actual brightness. The palette circuit 204 supplies, to a display section 207, image to be displayed.

5 [0009] When the conventional three-dimensional shape drawing device described in the patent document 2 draws a new three-dimensional shape, the MAXZ 211 and MINZ 212 of the block 213 of the screen 214 are compared with a Z value of the pixel to be newly drawn. If the Z value of the pixel to be newly drawn
10 is greater than the MAXZ 211, the three-dimensional shape drawing device 200 terminates drawing the pixel.

[0010] On the other hand, if the Z value of the pixel to be newly drawn is smaller than the MINZ 212, the three-dimensional shape drawing device 200 writes a new Z value in the Z main buffer
15 206, and writes a new pixel value in the frame buffer 203. Further, the three-dimensional shape drawing device 200 overwrites the MINZ 212 with the new Z value.

[0011] If the Z value of the pixel to be newly drawn is between the MINZ 212 and MAXZ 211, the three-dimensional shape drawing
20 device 200 refers to the Z main buffer 206, and compares the Z value of the pixel to be drawn with a Z value retained by the Z main buffer 206.

[0012] If the Z value of the pixel to be newly drawn is smaller than the Z value retained by the Z main buffer 206, the
25 three-dimensional shape drawing device 200 overwrites the Z value

retained by the Z main buffer 206 with a new Z value. Then, the three-dimensional shape drawing device 200 checks, among the Z values retained by the Z main buffer 206, all Z values belonging to the block 213, and then sets a greatest Z value among all the Z values belonging to the block 213 as the MAXZ 211.

[0013] As described above, the conventional three-dimensional shape drawing device described in the patent document 2 accesses the Z main buffer 206 only when the Z value of the pixel to be newly drawn is between the maximum Z value and the minimum Z value of the block in which the pixel is positioned. As a result, the frequency of accessing the Z main buffer 206 is reduced.

[Patent Document 1] Japanese Laid-Open Patent Publication No. 62-42281

[Patent Document 2] Japanese Laid-Open Patent Publication No. 8-161526 (pages 5-7, FIGS. 1 and 4)

DISCLOSURE OF THE INVENTION

PROBLEMS TO BE SOLVED BY THE INVENTION

[0014] However, while the conventional three-dimensional shape drawing device described in the patent document 2 is capable of eliminating the necessity to refer to the Z main buffer for every pixel, the conventional three-dimensional shape drawing device is still required to read all Z values within a block in order to determine a value to be the maximum Z value when the maximum Z value retained in the ZR buffer is updated. Moreover, when

performing a calculation to determine the maximum value, the conventional device cannot process a next pixel. This interrupts drawing processing of the device. As a result, the device cannot draw a three-dimensional shape at a high speed.

5 [0015] Therefore, an object of the present invention is to provide a three-dimensional shape drawing device and a three-dimensional shapedrawingmethodwhicharecapableofdrawing a three-dimensional shape at a high speed.

10 SOLUTION TO THE PROBLEMS

[0016] The present invention is a three-dimensional shape drawing device for drawing a three-dimensional shape by using Z buffer algorithm, the three-dimensional shape drawing device comprising: a depth value calculation section for calculating a
 15 depth value of a pixel to be drawn; a high order Z-buffer memory for retaining high order bits of a depth value of a pixel to be displayed as a front face, the depth value being among depth values having been calculated by the depth value calculation section; a low order Z-buffer memory for retaining low order bits of the
 20 depth value of the pixel to be displayed as the front face, the depth value being among the depth values having been calculated by the depth value calculation section; a high order bit comparing section for reading the high order bits retained by the high order Z-buffer memory and comparing the high order bits with high order
 25 bits of the depth value calculated by the depth value calculation

section; a low order bit comparing section for, when a result of a comparison performed by the high order bit comparing section shows that the high order bits of the depth value calculated by the depth value calculation section have a same value as that of the high order bits of the depth value retained by the high order Z-buffer memory, reading the low order bits of the depth value retained by the low order Z-buffer memory and comparing the low order bits with low order bits of the depth value calculated by the depth value calculation section; and a record update section for, when the result of the comparison performed by the high order bit comparing section shows that a depth indicated by the high order bits of the depth value calculated by the depth value calculation section is shallower than a depth indicated by the high order bits of the depth value retained by the high order Z-buffer memory, updating the high order bits of the depth value retained by the high order Z-buffer memory and the low order bits of the depth value retained by the low order Z-buffer memory by using the depth value calculated by the depth value calculation section, and for, when a result of a comparison performed by the low order bit comparing section shows that a depth indicated by the low order bits of the depth value calculated by the depth value calculation section is shallower than a depth indicated by the low order bits of the depth value retained by the low order Z-buffer memory, updating the low order bits retained by the low order Z-buffer memory by using the depth value calculated by the depth value

calculation section.

[0017] The present invention is also a three-dimensional shape drawing method for drawing a three-dimensional shape by using Z buffer algorithm, the three-dimensional shape drawing method comprising the steps of: calculating a depth value of a pixel to be drawn; reading, from a high order Z-buffer memory retaining high order bits of a depth value of a pixel to be displayed as a front face, the high order bits, the depth value being among depth values having been calculated at the step of calculating a depth value, and comparing the high order bits having been read with high order bits of the depth value calculated at the step of calculating a depth value; when the high order bits of the depth value calculated at the step of calculating a depth value are determined, at the step of comparing the high order bits, to have a same value as that of the high order bits of the depth value retained by the high order Z-buffer memory, reading, from a low order Z-buffer memory retaining low order bits of the depth value of the pixel to be displayed as the front face, the low order bits, the depth value being among the depth values having been calculated at the step of calculating a depth value, and comparing the low order bits having been read with low order bits of the depth value calculated at the step of calculating a depth value; and updating the high order bits of the depth value retained by the high order Z-buffer memory and the low order bits of the depth value retained by the low order Z-buffer memory by using the depth value calculated

by the step of calculating a depth value when a depth indicated by the high order bits of the depth value calculated at the step of calculating a depth value is determined, at the step of comparing the high order bits, to be shallower than a depth indicated by the high order bits of the depth value retained by the high order Z-buffer memory, and updating the low order bits retained by the low order Z-buffer memory by using the depth value calculated at the step of calculating a depth value when a depth indicated by the low order bits of the depth value calculated at the step of calculating a depth value is determined, at the step of comparing the low order bits, to be shallower than a depth indicated by the low order bits of the depth value retained by the low order Z-buffer memory.

15 EFFECT OF THE INVENTION

[0018] According to the present invention, a three-dimensional shape drawing device and a three-dimensional shape drawing method are provided which are capable of drawing a three-dimensional shape at a high speed.

20 BRIEF DESCRIPTION OF THE DRAWINGS

[0019] [FIG. 1] FIG. 1 is a functional block diagram showing a configuration of a three-dimensional shape drawing device according to an embodiment of the present invention.

25 [FIG. 2] FIG. 2 is a diagram schematically showing a

depth value retained by a high order Z-buffer memory 102 and a low order Z-buffer memory 104.

[FIG. 3] FIG. 3 is a flowchart showing operations of the three-dimensional shape drawing device shown in FIG. 1.

5 [FIG. 4] FIG. 4 is a flowchart showing a detailed process at subroutine step S19 shown in FIG. 3.

[FIG. 5] FIG. 5 is a block diagram showing an exemplary hardware configuration of the three-dimensional shape drawing device.

10 [FIG. 6] FIG. 6 is a block diagram showing an exemplary hardware configuration of the three-dimensional shape drawing device.

[FIG. 7] FIG. 7 is a block diagram showing a configuration of a three-dimensional shape drawing device 200
15 described in a patent document 2.

[FIG. 8] FIG. 8 schematically shows a screen 214 for displaying a three-dimensional shape drawn by the three-dimensional shape drawing device of FIG. 7 and a configuration of a ZR buffer 205.

20

DESCRIPTION OF THE REFERENCE CHARACTERS

[0020] 101 drawing section
102 high order Z-buffer memory
103 image memory
25 104 low order Z-buffer memory

	105	high order bit comparing section
	106	low order bit comparing section
	107	high order Z-buffer clearing section
	108	low order Z-buffer clearing section]
5	110	depth value calculation section
	111	brightness and material calculation section
	121	CPU
	122	high-speed graphics memory
	123	low-speed graphics memory
10	201	image composing section
	202	drawing processing section
	203	frame buffer
	204	palette circuit
	205	ZR buffer
15	206	Z main buffer
	207	display section
	208	image supply section
	209	pixel drawing section
	210	drawing determination section
20	211	maximum Z value within a block
	212	minimum Z value within a block
	213	example of a block resulting from dividing a screen
	214	screen
	401, 402	figures
25	403	pixel comprising a figure

BEST MODE FOR CARRYING OUT THE INVENTION

[0021] Hereinafter, an embodiment of the present invention will be described with reference to the drawings.

5 [0022] FIG. 1 is a functional block diagram showing a configuration of a three-dimensional shape drawing device 10 according to the embodiment of the present invention. In FIG. 1, the three-dimensional shape drawing device 10 comprises a drawing section 101, a high order Z-buffer memory 102, an image
10 memory 103, a low order Z-buffer memory 104, a high order bit comparing section 105, a low order bit comparing section 106, a high order Z-buffer clearing section 107 and a low order Z-buffer clearing section 108.

[0023] The image memory 103 stores data of an image which has
15 been drawn by the drawing section 101, the image whose hidden surface having been removed. The high order Z-buffer memory 102 stores a depth value of a pixel to be displayed as a front face of the image. To be specific, the high order Z-buffer memory 102 stores high order bits of a depth value of each pixel stored in the image
20 memory 103. The low order Z-buffer memory 104 also stores the depth value of the pixel to be displayed as the front face. To be specific, the low order Z-buffer memory 104 stores low order bits of the depth value of the each pixel stored in the image memory 103. For example, in the case where the depth value is represented
25 as 24 bits, the high order Z-buffer memory 102 retains high order

8 bits of the depth value, and the low order Z-buffer memory 104 retains low order 16 bits of the depth value.

[0024] Controlled by a timing control section 109 is a timing of drawing a figure to show a next display when a previous display is completed. To be specific, at a timing when the figure should be drawn for the next display, the timing control section 109 instructs the high order Z-buffer clearing section 107 and the low order Z-buffer memory 104 to initialize the image memory 103, the high order Z-buffer memory 102 and the low order Z-buffer memory 104. Also, the timing control section 109 instructs the drawing section 101 to start drawing the figure at the timing when the figure should be drawn for the next display.

[0025] In response to the instruction from the timing control section 109, the high order Z-buffer clearing section 107 initializes the high order Z-buffer memory 102 and the image memory 103. Also, the low order Z-buffer clearing section 108 initializes the low order Z-buffer memory 104 in response to the instruction from the timing control section 109.

[0026] The high order bit comparing section 105 compares high order bits of a depth value received from a later-described depth value calculation section 110 with high order bits of a depth value read from the high order Z-buffer memory 102. The low order bits which the high order bit comparing section 105 reads from the high order Z-buffer memory 102 are low order bits of a depth value of a pixel which is in a same position as that of a pixel which the

depth value calculation section 110 has used for calculation. Then, the high order bit comparing section 105 notifies a comparison result of the two sets of high order bits to a later-described read control section 112.

5 [0027] The low order bit comparing section 106 compares low order bits of the depth value received from the later-described depth value calculation section 110 with low order bits of the depth value read from the low order Z-buffer memory 104. The low order bits which the low order bit comparing section 106 reads
10 from the low order Z-buffer memory 104 are low order bits of the depth value of the pixel which is in the same position as that of the pixel which the depth value calculation section 110 has used for the calculation. Then, the low order bit comparing section 106 notifies a comparison result of the two sets of low order bits
15 to the later-described read control section 112.

[0028] The drawing section 101 includes the depth value calculation section 110, a brightness and material calculation section 111, a read control section 112 and a write control section 113. The drawing section 101 converts polygon data into pixel
20 data.

[0029] The depth value calculation section 110 calculates a depth value of a pixel. Then, the depth value calculation section 110 sends the calculated depth value to the high order bit comparing section 105, the low order bit comparing section 106 and the write
25 control section 113. Further, the depth value calculation section

110 generates a calculation notification signal for notifying that the depth value has been calculated, and sends the signal to the read control section 112. Note that, high order bits and low order bits are sent, together with positional information indicating a position of the depth value, to the high order bit comparing section 105 and the low order bit comparing section 106, respectively. The positional information may be, for example, coordinates. Hereinafter in the present embodiment, unless it is described differently, the depth value calculation section 110 sends the calculated depth value together with the positional information to the high order bit comparing section 105, the low order bit comparing section 106 and the control section 113.

[0030] Upon receiving the calculation notification signal, the read control section 112 instructs the high order bit comparing section 105 to compare the high order bits of the depth value calculated by the depth value calculation section 110 with the high order bits of the depth value retained by the high order Z-buffer memory 102. When a result of the comparison of the two sets of high order bits performed by the high order bit comparing section 105 shows that the high order bits of the depth value calculated by the depth value calculation section 110 are same as the high order bits of the depth value retained by the high order Z-buffer memory 102, the read control section 112 instructs the low order bit comparing section 106 to compare the low order bits of the depth value calculated by the depth value calculation section 110

with the low order bits of the depth value retained by the low order Z-buffer memory 104.

[0031] Based on notifications from the high order bit comparing section 105 and low order bit comparing section 106, the read control section 112 determines whether or not to update the depth value retained by the high order Z-buffer memory 102 and the low order Z-buffer memory 104. When a result of the comparison performed by the high order bit comparing section 105 shows that a depth indicated by the high order bits of the depth value calculated by the depth value calculation section 110 is shallower than a depth indicated by the high order bits of the depth value retained by the high order Z-buffer memory 102, the read control section 112 instructs a write control section 113 to update, with the high order bits and low order bits of the depth value calculated by the depth value calculation section 110, the high order bits of the depth value retained by the high order Z-buffer memory 102 and the low order bits retained by the low order Z-buffer memory 104.

[0032] Also, when a result of the comparison performed by the low order bit comparing section 106 shows that a depth indicated by the low order bits of the depth value calculated by the depth value calculation section 110 is shallower than a depth indicated by the low order bits of the depth value retained by the low order Z-buffer memory 104, the read control section 112 instructs the write control section 113 to update the low order bits retained

by the low order Z-buffer memory 104 with the depth value calculated by the depth value calculation section 110.

[0033] In response to the instruction from the read control section 112, the write control section 113 updates either only the low order bits retained by the low order Z-buffer memory 104 or both the high order bits retained by the high order Z-buffer memory 102 and the low order bits retained by the low order Z-buffer memory 104. When the write control section 113 updates such a record retained by the high order Z-buffer memory 102 or the low order Z-buffer memory 104, the write control section 113 instructs the brightness and material calculation section 111 to calculate a brightness value and material value and writes these values in the image memory 103.

[0034] In response to the instruction from the write control section 113, the brightness and material calculation section 111 calculates, e.g., a brightness value indicating brightness of a pixel and a material value indicating a texture of a surface of the pixel. Then, the brightness and material calculation section 111 writes the calculated brightness value and material value in the image memory 103.

[0035] FIGS. 2 and 3 are flow charts showing operations of the three-dimensional shape drawing device 10 shown in FIG. 1. First, when a displaying process of a previous display is completed, the timing control section 109 instructs the high order Z-buffer clearing section 107 to initialize the image memory and the high

order Z-buffer memory 102. Then, the high order Z-buffer clearing section 107 initializes the high order Z-buffer memory 102 and the image memory 103, and overwrites, with an initial value, values retained by the high order Z-buffer memory 102 and the image memory (Step S11). The initial value is, for example, a depth value indicating a farthest point from a viewpoint. Hereinafter, an example will be described in which a depth value is represented in hexadecimal, and the depth value indicating the farthest point is 000000H in 24 bits. For example, when high order bits are 8 bits, the high order Z-buffer clearing section 107 initializes the high order Z-buffer memory 102 by using 00H at step S11.

[0036] Next, the timing control section 109 instructs the low order Z-buffer clearing section 108 to initialize the low order Z-buffer memory 104. The low order Z-buffer clearing section 108 then initializes the low order Z-buffer memory 104, and overwrites a value retained by the low order Z-buffer memory 104 with the initial value (step S12). For example, when low order bits are 16 bits, the low order Z-buffer clearing section 108 initializes the low order Z-buffer memory 104 by using 0000H at step S12.

[0037] The depth value calculation section 110 calculates a depth value Z_{now} of each of pixels comprising a three-dimensional shape outputted from an image supply section (not shown in the figure), and sends the calculated value Z_{now} to the high order bit comparing section 105, the low order bit comparing section 106 and the write control section 113. The depth value calculation

section 110 also sends the calculation notification signal to the read control section 112. The read control section 112 instructs the high order bit comparing section 105 to compare high order bits of the depth value calculated by the depth value calculation section 110 with high order bits of a depth value retained by the high order Z-buffer memory 102.

[0038] Then, the high order bit comparing section 105 reads high order bits ZoldHI, i.e., the high order bits of the depth value retained by the high order Z-buffer memory 102 (step S13), and compares the high order bits ZoldHI with high order bits ZnowHI received from the depth value calculation section 110 (step S14). Then, the high order bit comparing section 105 notifies a result of the comparison to the read control section 112.

[0039] Upon receiving the result of the comparison performed by the high order bit comparing section 105, the read control section 112 determines based on the result of the comparison whether the high order bits of the depth value calculated by the depth value calculation section 110 are same as the high order bits of the depth value retained by the high order Z-buffer memory 102 (step S15). In the case where the two sets of high order bits are same, i.e., $Z_{nowHI} = Z_{oldHI}$, a positional relationship between pixels cannot be determined only based on the comparison of the high order bits. For this reason, the read control section 112 instructs the low order bit comparing section 106 to compare low order bits ZnowLo of the depth value calculated by the depth value calculation

section 110 with low order bits of the depth value retained by the low order Z-buffer memory 104.

[0040] The low order bit comparing section 106 reads low order bits ZoldLO, i.e., the low order bits of the depth value retained by the low order Z-buffer memory 104 (step S16), and compares the low order bits ZoldLO with the low order bits ZnowLO received from the depth value calculation section 110, and notifies a result of the comparison to the read control section 112 (step S17).

[0041] The read control section 112 determines whether or not a depth indicated by low order bits of a depth value of a pixel to be newly drawn is shallower than a depth indicated by low order bits retained by the low order Z-buffer memory 104 (step S18). If an inequality $Z_{nowLO} > Z_{oldLO}$ is satisfied, i.e., if the pixel to be newly drawn is positioned in front of a pixel having already been drawn, the read control section 112 proceeds to a process of step S21 of FIG. 3. If an inequality $Z_{nowLO} \leq Z_{oldLO}$ is satisfied, a process at the brightness and material calculation section 111 is not performed, and then the read control section 112 performs processes for a next pixel, i.e., proceeds to a process at step S13.

[0042] At step S21 of FIG. 3, the write control section 113 writes the low order bits ZnowLO in the low order Z-buffer memory 104.

[0043] Then, the write control section 113 instructs the brightness and material calculation section 111 to calculate a

brightness value and a material value. The brightness and material calculation section 111 then calculates values I_{now} of the pixel, e.g., a brightness value I_{now} , and writes the values in the image memory 103 (step S22).

5 [0044] The read control section 112 determines whether or not a polygon drawing on a screen has been completed (step S23). If the polygon drawing has not been completed, the read control section 112 returns to the process at step S13. If the polygon drawing has been completed, i.e., if there is no more image data to be
10 outputted from the image supply section, the read control section 112 terminates processing.

[0045] If the two sets of high order bits are not same at step S15, the read control section 112 determines whether or not a depth indicated by the high order bits of the depth value of the pixel
15 to be newly drawn is shallower than a depth indicated by the high order bits retained by the high order Z-buffer memory 102 (step S19). If an inequality $Z_{nowHI} < Z_{oldHI}$ is satisfied, i.e., if the pixel to be drawn next is positioned behind the pixel having already been drawn, the process at the brightness and material
20 calculation section 111 is not performed, and then the read control section 112 performs processes for a next pixel. In this case, the read control section 112 proceeds to the process at step S13.

[0046] On the other hand, if an inequality $Z_{nowHI} > Z_{oldHI}$ is satisfied, i.e., if the pixel to be newly drawn is positioned in
25 front of the pixel having already been drawn, the read control

section 112 instructs the write control section 113 to update the depth value retained by the high order Z-buffer memory 102 and the low order Z-buffer memory 104. The write control section 113 proceeds to a process at step S20 of FIG. 3, and writes the high order bits ZnowHI in the high order Z-buffer memory 102. Since processes performed after step S20 are same as those described above, descriptions thereof will be omitted.

[0047] FIG. 4 is a diagram schematically showing depth values of figures 401 and 402 retained by the high order Z-buffer memory 102 and the low order Z-buffer memory 104. In both the high order Z-buffer memory 102 and the low order Z-buffer memory 104, a pixel 403 is positioned at a same coordinate point. In order to simplify a description, an illustration showing the image memory 103 is omitted in FIG. 4.

[0048] In an example shown in FIG. 4, since the high order Z-buffer memory 102 retains high order 8 bits, and the low order Z-buffer memory 104 retains low order 16 bits, the high order Z-buffer clearing section 107 initializes a value retained by the high order Z-buffer memory 102 to 00H, and the low order Z-buffer clearing section 108 initializes a value retained by the low order Z-buffer memory 104 to 0000H. Note that, instead of having the high order Z-buffer clearing section 107 and the low order Z-buffer clearing section 108 which are dedicated to initializing the high order Z-buffer memory 102 and the low order Z-buffer clearing section 108, the drawing section 101 may have a function for

overwriting the values retained by the high order Z-buffer memory 102 and low order Z-buffer memory 104 with the value 000000H indicating the farthest point.

[0049] Next, a description will be given by taking the pixel 403 shown in FIG. 4 as an example. First, the high order Z-buffer clearing section 107 initializes the high order Z-buffer memory 102 with the initial value 00H, and the low order Z-buffer clearing section 108 initializes the low order Z-buffer memory 104 with the initial value 0000H. Then, when the figure 401 is drawn, the high order bit comparing section 105 compares high order 8 bits Z401HI of a depth value of a position of the pixel 403 of the figure 401 with the initial value 00H retained by the high order Z-buffer memory 102. A result of the comparison shows that Z401HI is greater than the initial value, i.e., the position indicated by Z401HI is in front of a position indicated by the initial value. Therefore, the write control section 113 writes Z401HI in the high order Z-buffer memory 102, and writes Z401LO in the low order Z-buffer memory 104. The brightness and material calculation section 111 calculates the brightness value and material value of the pixel 403, and writes the values in the image memory 103.

[0050] Next, when the figure 402 is drawn, the high order bit comparing section 105 compares high order 8 bits Z402HI of a depth value of the position of the pixel 403 of the figure 402 with Z401HI of the depth value of the position of the pixel 403 of the figure 401 retained by the high order Z buffer memory 102. If Z402HI

and Z401HI of the depth values are same, the read control section 112 instructs the low order bit comparing section 106 to compare low order bits of the depth values. Then, the low order bit comparing section 106 compares low order 16 bits Z402LO of the depth value of the position of the pixel 403 of the figure 402 with Z401LO retained by the low order Z-buffer memory 104. Since a result of the comparison is $Z401LO > Z402LO$, the values of the pixel 403 retained by the high order Z-buffer memory 102, the low order Z-buffer memory 104 and the image memory 103 are not updated.

10 Then, a process for drawing a next pixel is begun.

[0051] As described above, according to the present embodiment, when a pixel data is drawn, only high order bits of depth values are compared first. Low order bits of the depth values are compared only when the high order bits of the depth values are same. Since only the high order bits of the depth values are compared first, bus traffic is reduced. As a result, fast drawing of a three-dimensional shape is realized. Note that, the bus traffic is represented as a product of the number of accesses to the memory and a bus width.

20 [0052] According to the present embodiment, a bus bandwidth can be reduced. Consequently, the fast drawing of the three-dimensional shape can be performed even with a limited bus width. Further, rendering of a pixel is performed only after hidden surface removal is completed. This allows drawing substantial number of figures even with low arithmetic capability.

25

[0053] The present embodiment gives the example in which the brightness and material calculation section 111 performs calculation only when the calculation is necessary after the comparison of depth values is completed. However, the calculation
5 may be performed concurrently with the calculation performed by the depth value calculation section 110. Also, the processes at the depth value calculation section 110, the high order bit comparing section 105, the low order bit comparing section 106 and the brightness and material calculation section 111 may be
10 performed by pipeline processing.

[0054] It is described in the present embodiment that the depth value of a pixel is represented as 24 bits in hexadecimal format, and the initial value of the depth value is 000000H. However, the initial value is not limited thereto. For example, the high
15 order Z-buffer memory and the low order Z-buffer memory may be initialized by using 010000H as a value indicating the farthest point.

[0055] It is also described in the present embodiment that when drawing for the next display is begun, both the high order z-buffer
20 memory and the low order Z-buffer memory are initialized. However, only the high order Z-buffer memory may be initialized by using a predetermined initial value, e.g., 00H or 11H. In this case, high order bits of depth values to be first compared by the high order bit comparing section are always written in the high order
25 Z-buffer memory. At the same time, low order bits of the depth

values are written in the low order Z-buffer memory. Thus, the initialization of the low order Z-buffer memory performed by the low order Z-buffer clearing section can be skipped. As a result, memory traffic is reduced. This allows drawing substantial number of figures even with low arithmetic capability.

[0056] Note that, the present embodiment gives the example in which the high order bits of a depth value are 8 bits, and the low order bits of the depth value are 16 bits, i.e., 24 bits in total. However, the total bit count, the number of high order bits and the number of low order bits may be arbitrarily set. Here, it is preferred that the number of high order bits is equal to or smaller than the number of low order bits, in order to reduce the amount of data to be read from the high order Z-buffer memory.

[0057] In the present embodiment, when the compared two sets of low order bits are same, the low order bits retained by the low order Z buffer are not updated. Alternatively, when the compared two sets of low order bits are same, the brightness value and the material value retained by the image memory as well as the low order bits retained by the low order Z buffer may be updated.

[0058] It is also described in the present embodiment that the high order Z-buffer memory and the low order Z-buffer memory are two memories which are physically separable. However, one memory may function as both the high order Z-buffer memory and the low order Z-buffer memory. In this case, high order bits and low order bits are separately recorded at different addresses of one memory.

[0059] If it is determined that a depth indicated by high order bits of a depth value calculated by the depth value calculation section is shallower than a depth indicated by high order bits of a depth value retained by the high order Z-buffer memory, the high order bit comparing section may add a flag to the high order bits of the depth value calculated by the depth value calculation section, and if it is determined that a depth indicated by low order bits of the depth value calculated by the depth value calculation section is shallower than a depth indicated by low order bits of the depth value retained by the low order Z-buffer memory, the low order bit comparing section may add a flag to the low order bits of the depth value calculated by the depth value calculation section. Here, when the flag is added to the high order bits of the depth value calculated by the depth value calculation section, the write control section may update the high order bits of the depth value retained by the high order Z-buffer memory and the low order bits of the depth value retained by the low order Z-buffer memory, and when the flag is added to the low order bits of the depth value calculated by the depth value calculation section, the write control section may update either only the low order bits of the depth value retained by the low order Z-buffer memory or both the high order bits of the depth value retained by the high order Z-buffer memory and the low order bits of the depth value retained by the low order Z-buffer memory.

[0060] (example)

Next, an example of the present invention will be described. FIG. 5 is a block diagram showing a hardware configuration of the three-dimensional shape drawing device 10. The three-dimensional shape drawing device 10 comprises a CPU 121, a high-speed graphics memory 122 and a low-speed graphics memory 123, the CPU 121 implementing a function of the drawing section 101. Note that, functions of the high order bit comparing section 105, the low order bit comparing section 106, the high order Z-buffer clearing section 107, the low order Z-buffer clearing section 108 and the timing control section 109 are also implemented by the CPU. However, in FIG. 5, only the drawing section 101 is illustrated as a representative example.

[0061] Since a functional block diagram of the three-dimensional shape drawing device 10 of FIG. 5 is identical with FIG. 1 of the above-described embodiment, FIG. 1 will be used as a reference. Also, since operations of the three-dimensional shape drawing device 10 of FIG. 5 are identical with those illustrated in FIG. 2 of the above embodiment, FIG. 2 will be used as a reference.

[0062] The high-speed graphics memory 122 includes the high order Z-buffer memory 102, and is, for example, a SRAM (Static Random Access Memory) or an ASIC embedded memory. The low-speed graphics memory 123 includes the image memory 103 and low order Z-buffer memory 104, and is, for example, a DRAM (Dynamic Random Access Memory).

[0063] High-speed processing is realized by using the high-speed graphics memory being capable of high-speed processing as the high order Z-buffer memory which is accessed more frequently than the low order Z-buffer memory and image memory. Since the low order Z-buffer memory is only required to retain high order 8 bits of a depth value, a necessary size of the high-speed graphics memory can be kept small. Moreover, a processing speed of the three-dimensional shape drawing device can be enhanced with a minimum increase in cost, by including, in the low-speed graphics memory which has a large capacity but is low-speed, the low order Z-buffer memory and the image memory which are accessed less frequently than the high order Z-buffer memory.

[0064] In the case where a three-dimensional shape is drawn by using an arithmetic unit such as a DSP, calculations of brightness value and other values are performed when a record kept in the high order Z-buffer memory or the low order Z-buffer memory is updated. This allows the three-dimensional shape drawing device of the present invention to draw more three-dimensional shapes than a conventional three-dimensional shape drawing device within a same time period, even if the arithmetic unit has a same capability as that of an arithmetic unit of the conventional three-dimensional shape drawing device.

[0065] It is described in this example that the functions of the drawing section, the high order bit comparing section, the low order bit comparing section, the high order Z-buffer clearing

section, the low order Z-buffer clearing section and the timing control section are implemented by the CPU. However, the functions of these sections are not necessarily implemented by the CPU. A function of each section may be implemented by a LSI executing a program for the function.

[0066] Although it is described in this example that the image memory and the low order Z-buffer memory are embedded in the three-dimensional shape drawing device, these memories do not necessarily have to be embedded in the three-dimensional shape drawing device.

[0067] FIG. 6 is a block diagram showing a hardware configuration of the three-dimensional shape drawing device 10 in the case where the image memory 103 and the low order Z-buffer memory are external memories. In FIG. 6, the three-dimensional shape drawing device comprises the CPU 121, an embedded memory 124 and a memory I/F 125, the CPU 121 implementing the function of the drawing section 101.

[0068] The memory I/F 125 controls an access by the drawing section 101 to the image memory 103 and the low order Z-buffer memory 104 which are included in an external memory 126.

[0069] Since the image memory and the low order Z-buffer memory are accessed less frequently than the high order Z-buffer memory, the image memory and the low order Z-buffer memory may be provided as external memories. Thus, a size of an embedded memory of the three-dimensional shape drawing device can be reduced.

INDUSTRIAL APPLICABILITY

[0070] The present invention is useful as a three-dimensional shape drawing device and a three-dimensional shape drawing method
5 for drawing a three-dimensional shape at a high speed.